Wolfgang Killmann
T-Systems debis Systemhaus Information Security Services, Bonn

Priv.-Doz. Dr. Werner Schindler
Bundesamt für Sicherheit in der Informationstechnik (BSI) , Bonn

A proposal for:

Functionality classes and evaluation methodology for true (physical) random number generators[1]

Version 3.1
25.09.2001

Contents

[1] The authors wish to express their thanks for the numerous comments, suggestions and notes that have been incorporated into this document.

A. Motivation, aims and overview of contents

A.1 Motivation and aims: Although random numbers play an important role in numerous cryptographic applications, ITSEC and CC do not specify any uniform evaluation criteria for random numbers. This document describes the evaluation criteria for true (physical) random number generators. This paper is a counterpart to the mathematical basis of [AIS20].

A.2 Overview of contents: Chapter B describes the object of investigation. Chapter C introduces two functionality classes (P1, P2), providing reasons for this classification. Chapter D describes the tasks of the evaluator, insofar as they are relevant for investigating the TRNG, but makes no claim to present the requirements of the ITSEC and CC criteria in their entirety. Chapter E provides several detailed examples to explain the class-specific requirements.

A.3 Comment:

(i) These evaluation methods cannot be applied to random number generators whose noise source lies outside of the TOE (e.g. random keyboard entries by the user).
(ii) If applicants use a physical random number generator that cannot be assigned to functionality class P1 or P2 and if they are applying for a German IT security certificate, then BSI must be contacted.

B. Definitions and notation

B.1 Definitions: A *true (physical) random number generator* (abbreviated to TRNG) uses the noise signals from an internal physical noise source to generate random numbers. The values that result directly from the digitisation of analogue noise signals are referred to as *digitised noise signals* in the following. The term *internal random number* is used to refer to the values following mathematical post-processing (optional; see also C.2) of the digitised noise signal sequence. An *ideal random number generator* (theoretical!) generates independent random numbers that assume all possible values to the same probability. In the following, we understand *online tests* as statistical tests or – more precisely – a test specification applied during effective operation to the digitised noise signal sequence generated by the TRNG or to internal random numbers with the aim of verifying that the TRNG is functioning correctly. A conspicuous statistical feature detected by an online test leads to a *noise alarm* which in turn leads to the TRNG being stopped at least temporarily. We speak of *total failure (of the noise source)* if the digitised noise signal sequence is constant from this time on. Depending on the context, we understand the *entropy per bit* as the quotient

(entropy per digitised noise signal / width of the binary representation of a digitised noise signal) or (entropy per internal random number / number of bits in the binary representation of an internal random number).

C. Functionality classes

C.0 Reason for introducing functionality classes: A TRNG contains an internal physical noise source. It usually delivers an analogue signal that is digitised for further processing. The digitised noise signal can be transformed into an internal random number sequence by means of post-processing in order to improve the probability distribution of the digitised noise signal sequence. For good physical noise sources, post-processing is not necessary and the digitised noise signal can be transmitted directly to the output block. In this case, the sequence of internal random numbers corresponds to the digitised noise signal sequence. The output block synchronises the continuous or non-periodic generation of the internal random sequence with the calling of the (external) random number sequence. The noise source delivers the entropy of the output random number sequence that increases with every generated random number.

It must be clarified whether – or rather to what extent – a physical random number generator behaves like an ideal random number generator. In contrast to [AIS20], however, it is hardly possible to provide theoretical proofs. Instead, the assessment of a physical random number generator is essentially based on statistical tests. On the basis of different potential attack scenarios, various applications can place different requirements on the properties of the external, and therefore of course also the internal, random numbers. In order to take this circumstance into account, we will introduce two functionality classes (P1 and P2) in the following. With regard to the intended applications, classes P1 and P2 essentially correspond to classes K1 and K2 as well as K3 and K4 in [AIS20].

Roughly speaking, the P1 property requires the internal random numbers to be statistically inconspicuous. The P2-specific requirements should guarantee that they are practically impossible to determine even if the predecessors or successors are known. Depending on the maximum attack potential (specified here in the strength of mechanisms) attributed to a potential perpetrator, the TOE must itself recognise total failure or any interference that occurs in the noise source and may need to be able to resist systematic manipulation attempts.

Various examples are discussed in Chapter E.

C.1 The applicant must at least specify:

(i) The desired functionality class (P1, P2) with the strength of mechanisms (ITSEC) and functions (CC).

(iia) Information about the TRNG's structure and mode of functioning, together with the specification form and specification depth required for the evaluation level, must be provided in the detailed design from ITSEC E2 and in the low level design as of CC EAL 4 in accordance with ADV_LLD.1. For ITSEC E1, the applicant must state the TRNG's structure and mode of functioning as part of the proof of the strength of mechanisms in accordance with [JIL], Section 6.5.

(iib) As of ITSEC E3 under implementation or for CC EAL5 under ATE_DPT.2 tests: Low-level design, the applicant must supply proof of the statistical tests in line with the intended functionality class.

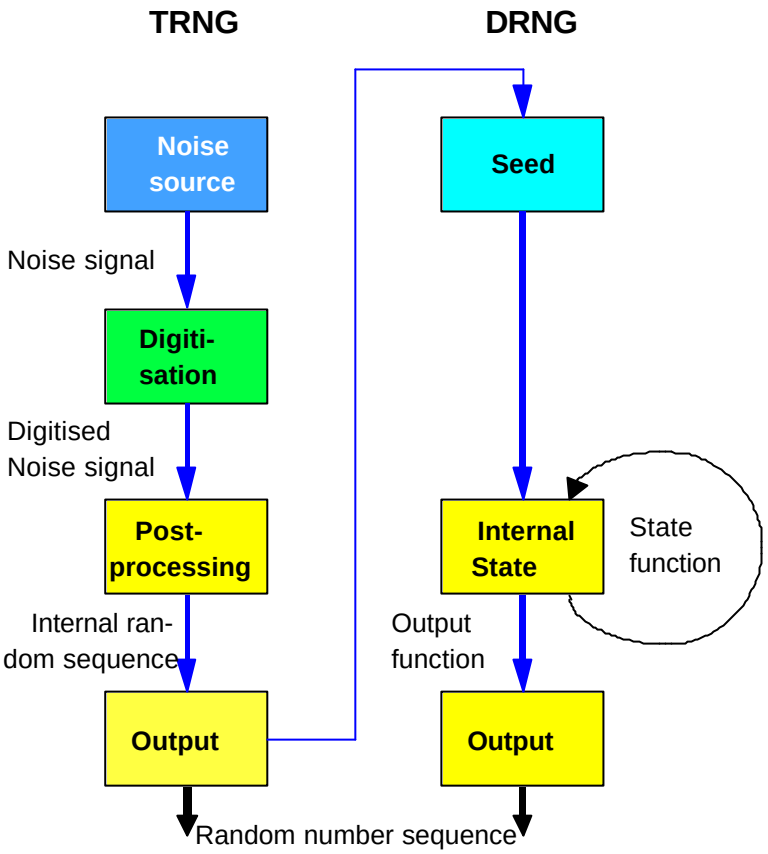
(iii) A clear description of how the noise signal is generated, together with an explanation of why a digitised random noise signal is to be induced in this way.

+ additional specifications listed in subsection f) of the corresponding functionality class.

C.2 Delimitation of the tagrget of evaluation TRNG:

A deterministic random number generator (DRNG) is given a seed by an external source and uses the state function to calculate a sequence of internal states. An image of this sequence generated using the output function is output (random number sequence). The overall entropy of the output sequence lies in the initial value. The overall entropy of a sequence of internal random numbers generated by a TRNG, on the other hand, increases with each random number. TRNGs are based on physical random processes, the observed analogue variables of which are prepared for digital processing. Processes that are digitised in all their parameters (time, level, etc.), i.e. limited to a finite number of states, will generally have deterministic behaviour and be regarded as DRNGs.

The following diagram visualises the essential parts of TRNGs and DRNGs as well as seed generation for DRNGs as a possible application for TRNGs. It represents the typical sequential processing of the signals. Network structures, for example a mixture of different analogue noise sources and post-worked signals that are already digitised, are basically possible but render the analysis more complicated and costly (e.g. decomposition). Mathematical post-processing of the digitised noise signals is optional. If it is not performed, the digitised noise signals agree with the internal random numbers.



A TOE can contain a random number generator as a combination of a TRNG for generating the seed and a DRNG for generating the random number sequence. In such a case, the analysis of the TRNG serves to back up the reason requested from the applicant in [AIS20], C.1(iv) that the seed generation really does induce the distribution p_A. The DRNG must be evaluated in accordance with [AIS20].

C.3 General comment on the specification of the functionality classes:

Sub-section d) describes the class-specific requirements. The details needed for evaluation in addition to C.1 (i) – (iii) are combined in sub-section f). The remaining sub-sections illustrate and justify the selection and aim of these requirements. Sub-sections i) and j) (see Chapter D) describe and explain the tasks of the evaluator.

Class P1

P1.a) Qualitative-intuitive description of the P1-specific requirements:

A sequence of random vectors formed from the internal random numbers $r_1,r_2,\ldots$ is most likely pairwise different. Internal random number sequences $r_1,r_2,\ldots$ and their projections to individual bits pass certain statistical tests (evaluation tests). If the strength of mechanisms or functions is ”medium” or “high”, total failure of the noise source should be detected when the TRNG is switched on and during operation. The statistical properties of the internal random numbers are tested during operation (”online tests”). If the strength of mechanisms or functions is ”high”, the statistical properties of the internal random numbers should not be influenced by external conditions (temperature, climate, ageing).

P1.b) Possible applications:

--- Challenge-response protocols

--- Openly transmitted, non-constant initialisation vectors

--- Seed generation for DRNGs of the classes K1 and K2 ([AIS20])

P1.c) Aim:

The statistical behaviour of the internal random numbers should be inconspicuous. This shall prevent replay and correlation attacks against cryptographic algorithms and protocols that are based on statistical weaknesses in the external random numbers used.

P1.d) Requirements on P1-TRNGs:

P1.d)(i) Random vectors formed from internal random number sequences pass the disjointness test T0. The test procedure and evaluation rules are specified in P1.i)(i).

P1.d)(ii) Taken as a binary string, internal random number sequences $r_1,r_2,\ldots$ and their projections to individual bits pass certain statistical tests. The evaluation rules are specified in sub-section P1.i)(ii).

P1.d)(iii) If the strength of mechanisms or functions is “medium” or “high”: If total failure of the noise source occurs when the TRNG is switched on, this must be detected immediately. In other words, external random numbers must not be output.

P1.d)(iv) If the strength of mechanisms or functions is “medium” or “high”: If total failure of the noise source occurs while the TRNG is being operated, it has

to be prohibited that random numbers are output whose internal random sequence was generated completely after the total failure. As an alternative, it is sufficient if, following total failure of the noise source, for each constant noise signal sequence the TRNG behaves like a K2-DRNG as defined by [AIS20], whose output sequence complies with the intended usage.

P1.d)(v) If the strength of mechanisms or functions is "high": The properties required in (i) and (ii) must be verified under the intended external usage conditions (temperature, power supply, etc.) insofar as these can influence the function of the noise source.

P1.d)(vi) If the strength of mechanisms or functions is "medium" or "high": In order for the TRNG to be operated, an online test must be implemented that checks the quality of the internal random numbers when triggered externally. It does not have to be possible for the online test to be triggered externally if, at the instigation of the TOE, all generated internal random numbers are tested using this online test or the online test is at least applied at regular intervals. For an ideal random number generator, the probability that at least one noise alarm will occur in the course of one year of typical use of the TRNG should be $\geq 10^{-6}$. As an alternative, it is permissible to use the online test to check the digitised noise signal sequence rather than the internal random numbers. In this case, however, it must be ensured that the mathematical post-processing does not reduce the average entropy per bit. (This will actually be the standard case if class P2 is desired because this requires the online tests to check the digitised noise signal sequence.)

Comment 1: The tot test as per P1d)(iv) and the online test as per P1.d)(vi) are usually implemented as a component of the TOE or can be implemented as external security measures in exceptional circumstances with good reason.

P1.e) Reason:

The external random numbers should render the cryptographic mechanisms in which they are used resistant to replay and correlation attacks. It does not matter for P1-TRNGs whether they can be determined or guessed with knowledge of external random number sub-sequences. In order to ensure that this goal is achieved, the statistical behaviour of the external random numbers should be inconspicuous. In other words, they should have similar statistical properties as if they had been generated by an ideal random number generator. As the external random numbers are usually formed by concatenating internal random numbers, it is required for P1-TRNGs that the statistical behaviour of the internal random numbers be inconspicuous. The internal random numbers are subject to various statistical tests (see P1.d) sections (i), (ii), (v), (vi)).

On P1.d)(i): The ”disjointness criterion” P1.d)(i) is a simple test for disproving the suitability of a TRNG for generating pairwise different external random numbers (→ Challenge-response protocols) for unsuitable TRNGs. Since it is not particularly powerful, this property is also checked by other statistical tests P1.d)(ii).

On P1.d)(ii): Although they are not particularly powerful, the statistical tests should be sufficiently powerful to prevent known attacks against the cryptographic algorithms that are based on the statistical weaknesses of the external random numbers. By projecting the internal random numbers onto the individual components, the individual random number bits are tested for likeness (see [AIS20], K1.e) and example E.2).

On P1.d)(iii): Following total failure of the noise source, the digitised noise signal, that is the input to mathematical post-processing, assumes a constant value. In particular, this means that, following total failure of the noise source, the same sequence of internal random numbers is generated each time the TRNG is restarted insofar as any registers belonging to the mathematical post-processing assume defined values upon each restart.

On P1.d)(iv): If the noise source fails completely, the digitised noise signal sequence becomes constant. If mathematical post-processing is present, it may behave like a DRNG as defined by [AIS20] following total failure of the noise source. After total failure has occurred, the contents of all registers is to be interpreted as a seed and the now constant input is to be interpreted as part of the algorithm for renewing the internal state of the DRNG. If this DRNG belongs to at least class K2 from [AIS20] for each constant digitised noise signal sequence, then even the external random numbers generated following the total failure of the noise source may be sufficient for the intended use.

On P1.d)(v): The function of physical noise sources may depend on the external usage conditions (which are described in the security specifications and operating documentation) or may be influenced by external interfaces of the TOE (which are described in the architectural design). In this case, it must be proved that the TRNG works properly under various usage conditions. This should also prevent targeted external attacks on the noise source that are directed at degrading the quality of the internal random numbers generated.

On P1.d)(vi): A noise alarm would occur occasionally even for an ideal random number generator. If this probability is too large, these would lead to too many (unnecessary!) shutdowns of the ideal random number generator. Depending on the ”defect”, that is the ”distance” from an ideal random number generator (more precisely: the deviation of the distribution of internal random numbers from independent and uniformly distributed random variables), this probability will usually

be greater for a real TRNG. If the noise alarm probability for the ideal random number generator is extremely small, even TRNGs that generate internal random numbers with relatively high statistical defects (e.g. non-uniform distribution or high dependencies of the internal random numbers) are not very likely to be detected by the online tests. For this reason, P1.d)(vi) stipulates a minimum probability for a noise alarm from an ideal noise source. Trivially, this probability increases with the number of online tests performed. Unsuitable mathematical post-processings, i.e. those that reduce the average entropy per bit, can transform good digitised noise signals into weak internal random numbers. If the online tests are applied to the digitised noise signals, therefore, proof must be provided that the mathematical post-processing does not reduce the average entropy per bit. This will actually be the standard case if class P2 is desired because P2.d)(xi) requires the online tests to be applied to the digitised noise signal sequence. In any case, it must be shown that the mathematical post-processing does not reduce the average entropy per bit (P2.d)(viii)).

P1.f) to be specified by the applicant in addition to C.1(i)-(iii):

P1.f)(iv) (Required if the strength of mechanisms or functions is „medium“ or „high“): The reasoning why P1.d)(iii) is met.

P1.f)(v) (Required if the strength of mechanisms or functions is „medium“ or „high“): The reasoning why a total failure of the noise source is detected sufficiently quickly while the TRNG is being operated (see P1.d)(iv)). If this is not ensured, a DRNG evaluation of the mathematical post-processing is required. In particular, the applicant must specify the parameters M, c and ε for which the mathematical post-processing is to be attributed the K1-specific property. The choice of parameters must be justified with regard to the intended uses of the TRNG.

P1.f)(vi) (Required if the strength of mechanisms or functions is „medium“ or „high“): The reasoning why P1.d)(iv) is met. Moreover, the consequences of the noise alarm must be described (shutdown of the noise source, intensive tests on the noise source, logging, etc.). If the noise source is taken back into operation following a noise alarm, it must be ensured that the internal random numbers do not have any unacceptable statistical weaknesses.

Comment 2: C.1 (iib) already gives rise to the obligation of the applicant to prove that requirements P1.d)(i), P1.d)(ii) and P.1d)(v) were met in the tests performed by the applicant, and to submit the test results. If the tot tests as per P1d)(iii) and (iv) as well as the online test as per P1.d)(vi) are to be implemented as external security measures, the applicant must submit a specification and reference implementation. Proof in accordance with P1.f)(iv) – (vi) can be provided based on the reference implementation.

P1.g) Explanations: A mechanism for detecting total failure of the physical noise source is referred to as a *tot test* in the following („tot“ stands for „total failure“) . This can be a suitable statistical test of the digitised noise sequence or of the internal random numbers. However, it is also possible to verify whether the digitised noise signal sequence is constant or whether the first n bits, n>15, are repeated following activation of the TRNG during operation (see Continuous random number generator test, [FI140-1], Section 4.11.2). The parameter n should be selected sufficiently large that a noise alarm triggered by the tot test will probably never occur in the ”lifecycle” of the TRNG, provided that the noise source has not actually failed completely.

P1.h) Examples: Total failure of the noise source / tot test: E.1, E.5, E.6, E.7;

Startup test: E.5, E.7;

Online test: E.6, E.7.

Class P2

P2.a) Qualitative-intuitive description of the P2-specific requirements:

The statistical behaviour of the digitised noise signal sequence is inconspicuous. If the strength of mechanisms or functions is medium or “high”, the functionality of the physical noise source is tested when the TRNG is switched on. Total failure of the physical noise source is detected when the TRNG is switched on or during operation. The TRNG tests the statistical properties of the digitised noise signals during operation at least when the tests are triggered externally (”online tests”). If the strength of mechanisms or functions is ”high”, the TOE must independently trigger execution of the online tests.

P2.b) Possible applications:

--- Generation of signature key pairs

--- Generation of DSS signatures (private key x or random number k; see [FI186])

--- Generation of session keys for symmetric encryption mechanisms

--- Random padding bits

--- Zero-knowledge proofs

--- Generation of seeds for DRNGs in classes K3 and K4

P2.c) Aims:

In addition to the P1-specific aim P1.c), the prospects of success for systematic guessing of the external random numbers (realised through systematic exhaustion attacks) – even if external random number sub-sequences are known – should at best

be negligibly higher than would be the case if the external random numbers had been generated by an ideal random number generator.

2.d) Requirements on P2-TRNGs:

The TRNG belongs to class P1 with at least the same strength of mechanisms and functions (downward compatibility).

P2.d)(vii) Digitised noise signal sequences meet particular criteria or pass statistical tests intended to rule out features such as multi-step dependencies. Moreover, the entropy test T8 is passed. The tests and evaluation rules are specified in sub-section P2.i).

Under certain conditions, alternative criteria can be used rather than the criteria or statistical tests specified under P2.i) (see ”Alternative criteria for P2.d)(vii); type 1” and ”Alternative criteria for P2.d); type 2”).

P2.d)(viii) If mathematical post-processing is present, it shall not reduce the average entropy per bit.

P2.d)(ix) If the strength of mechanisms or functions is “medium” or “high”, statistical minimum properties of the digitised noise signal sequence must be proved each time the TRNG is started. Random numbers must not be output before the statistical tests are completed.

P2.d)(x) If the strength of mechanisms or functions is “medium” or “high”: If total failure of the noise source occurs while the TRNG is in operation, it has to be prohibited that random numbers are output whose corresponding internal random sequence was generated completely after the total failure.

P2.d)(xi) If the strength of mechanisms or functions is “medium” or “high”: For the TRNG to be operated, an online test must be implemented with which the statistical quality of the digitised noise signal sequence can be checked. It must be possible to trigger this online test externally or the TRNG must trigger the online test itself. The latter must happen continuously or at least at regular intervals. The online test itself and the call schema must be suitable for detecting unacceptable statistical defects or deterioration of the statistical properties of the digitised noise signal sequence within an acceptable period of time. For an ideal random number generator, the probability that at least one noise alarm will occur in the course of one year of typical use of the TRNG should be $\geq 10^{-6}$.

P2.d)(xii) If the strength of mechanisms or functions is “high”: The properties required in P2.d)(vii) must be verified under the intended external usage conditions (temperature, power supply, etc.) insofar as these can influence the function of the noise source.

P2.d)(xiii) If the strength of mechanisms or functions is “high”: The TRNG must trigger the online test itself.

Comment 1: The tot test as per P1d)(iv) and P2.d)(x), the startup test as per P2.d)(ix) and the online test as per P1.d)(vi), P2.d)(xi) and P2.d)(xiii) are usually implemented as a component of the TOE. They can be implemented as external security measures in exceptional circumstances with good reason.

Alternative criteria for P2.d)(vii); type 1: The aim of P2.d)(vii) is to guarantee P2.c) for selected prototypes by verifying a minimum entropy limit for each internal random bit with a negligibly small error probability. If the digitised noise signal sequence does not meet criterion P2.d)(vii), the applicant may alternatively submit the following proof:

- Internal random number sequences pass the statistical tests specified in P2.i)(vii).
- *Clear* proof that the internal random numbers achieve the goal set with criterion P2.d)(vii). The proof must be provided taking into account the mathematical post-processing and on the basis of the empirical properties of the digitised noise signal sequence.

It is then conceded that the TRNG meets an alternative criterion equivalent to P2.d)(vii). The "comprehensible proof" mentioned in the second point can be based on statistical tests of the internal random numbers in as much as their suitability is justified.

Alternative criteria for P2.d)(vii); type 2: The aim of P2.d)(vii) is to guarantee P2.c) for selected prototypes by verifying a minimum entropy limit for each internal random bit with a negligibly small error probability. If the statistical tests required to prove property P2.d)(vii) (see P2.i)(vii)) cannot be applied to the noise signal sequence, the applicant may alternatively submit the following proof:

- Internal random number sequences pass the statistical tests specified in P2.i)(vii).
- *Comprehensible and plausible description* of a mathematical model of the physical noise source and the statistical properties of the digitised noise signal sequence derived from it.
- Specification of statistical tests that guarantee the goal defined in criterion P2.d)(vii) insofar as the internal random numbers pass these tests. It shall be comprehensibly justified that these tests are suitable. The proof must be provided taking into account the mathematical post-processing and on the basis of the statistical properties of the noise signal sequence derived from the mathematical model of the noise source.

It is then conceded that the TRNG meets an alternative criterion equivalent to P2.d)(vii). Alternative proof of criterion P2.d)(vii) in accordance with the ”Alternative criteria for P2.d)(vii); type 2” is considerably more difficult and extensive than

alternative proof in accordance with ”Alternative criteria for P2.d)(vii); type 1”. It will only be possible in exceptional situations.

Comment 2: (on P2.(ix) and P2.(xi)): If the internal random numbers rather than the digitised noise sequence are tested when the TRNG is started or through the online tests, then the applicant must provide separate justification of the effectiveness of these tests.

P2.e) Reason:

The P2-specific aim is guaranteed if the average entropy increase per internal random number – and thus also the average entropy increase per external random number – is close to the values of ideal random number generators.

On P2.d)(vii): The one-dimensional distribution of the digitised noise signal sequence should not differ too greatly from the uniform distribution. Differences in the one-step transition probabilities for the different predecessors are tolerated to a certain extent. However, the digitised noise signal sequence should not have any multi-step dependencies. The entropy of the digitised noise signal sequence is a measure for the randomness obtained from the noise source. The average entropy increase per digitised noise signal should therefore not fall below a minimum amount. Empirically, however, the entropy of a random number sequence can only be estimated reliably under certain model assumptions regarding the underlying probability distribution (independent, Markovian, finite memory, etc.). In fact, the expected value for test variable T8 is equal to the entropy per L-bit block if the bit sequence to be tested is generated by an independent, stationary binary-value noise source (see Chapter F, Test T8). The requirements on the empirical distribution in P2.i)(vii) shall therefore, amongst other things, guarantee that the entropy estimate is reliable. (The tests defined in P2.i)(vii.b)-P2.i)(vii.d) tolerate only low one-step and negligible two/three-step transition probabilities.) If we look at fairly short periods of time, it should be realistic to assume that the noise source is stationary. If the tests specified in P2.i)(vii) are passed, we can conclude (although this is not mathematically proven!) that the digitised noise signal sequence has a high degree of entropy.

In general, the digitised noise signal sequence will probably be skewed (i.e. not uniformly distributed) and may have one-step or even multi-step dependencies but certainly no complicated algebraic dependencies. Post-processing should reduce these weaknesses in the digitised noise signal sequence and prevent negative effects on the output values. However, some mathematical post-processing does not reduce weaknesses in the digitised noise signal sequence but merely blurs them or transforms them into other weaknesses, which frequently renders the application of common statistical tests on the internal random numbers ineffective (see example E.1). For this reason, the P2-specific tests should be applied to the digitised noise signal sequence

whenever possible. If the tests are applied to the internal random sequence, proof must be provided that they are useable and effective (see comment 2).

On P2.d)(viii): Mathematical post-processing should not reduce the entropy of the digitised noise signal sequence in any way.

On P2.d)(ix): This should ensure that serious weaknesses in the digitised noise signal sequence are detected before any random numbers are output. This in particular covers requirement P1.d)(iii).

On P2.d)(xi) and P2.d)(xii): For justification of a minimum probability of a noise alarm for an ideal noise source, see the reasoning for P1.d(v). Based on component tolerances it is possible for noise sources of the same type to generate digitised noise signal sequences with different statistical properties. Furthermore, the properties of the components may change over time (ageing effects). It is the task of online tests to detect both phenomena. Deviations in the digitised noise signal sequences from uniform distribution and independency ($\rightarrow$ ideal random number generator) are unavoidable and can be tolerated up to a certain extent. In the case of unacceptable deviations, the online test should give a noise alarm as quickly as possible. As each noise alarm causes the TRNG to be shut down at least temporarily, the online test should, on the other hand, not give rise to an overly large probability of a noise alarm for tolerable deviations.

P2.f) <u>to be specified by the applicant in addition to C.1(i)-(iii) and P1.f)(iv)-(vi):</u>

As of evaluation level E2, C1.(ii.b) requires the applicant to provide proof that the statistical tests described in P2.i)(vii) have been performed and to submit the test results. If the digitised noise signal sequence does not meet criterion P2.d)(vii) or if the digitised noise signal sequence cannot be tested, the applicant must specify alternative mechanisms and demonstrate their effectiveness. (See P2.d) ”Alternative criteria for P2.d)(vii); type 1” and ”Alternative criteria for P2.d)(vii); type 2”.)

P2.f)(vii) Justification that P2.d)(viii) is met.

P2.f)(viii) Justification that P2.d)(ix) is met.

P2.f)(ix) Proof that P2.d)(x) is met.

P2.f)(x) Proof and justification that P2.d)(xi) is met (possibly taking into account P2.d), comment 2). It is necessary to at least give an approximation of the probability of a noise alarm for certain (tolerated / not tolerated) deviations in the digitised noise signal sequence from the ideal behaviour (uniform distribution, independence $\rightarrow$ ideal random number generator). Applicants must specify and justify which deviations are considered to be

acceptable. The mathematical post-processing can be used in the justification.

P2.f)(xi) Proof that the tests specified in P2.i)(xii) have been performed and provision of the test results.

P2.f)(xii) Proof that P2.d)(xiii) is met. Moreover, the consequences of the noise alarm must be described (shutdown of the noise source, intensive tests on the noise source, logging, etc.). If the noise source is operated again following a noise alarm, it must be ensured that the digitised noise signals do not have any unacceptable statistical weaknesses.

Comment 3: C1.(ii.b) already gives rise to the obligation of the applicant to prove that requirements P2.d)(vii) were met in the tests performed by the applicant, and to provide the test results. If the tot test as per P1d)(iv) and P2.d)(x), the startup test as per P2.d)(ix), as well as the online test as per P1.d)(vi), P2.d)(xi) and P2.d)(xiii) are to be implemented as external security measures, the applicant must submit a specification and reference implementation for this purpose. Proof in accordance with P1.f)(v), P1.f)(vi), P2.f)(viii), P2.f)(ix), P2.f)(x) and P2.f)(xii) can be provided in a reference implementation.

P2.g) Explanations: A P2 evaluation is not possible if the strength of mechanisms or functions is low.

P2.h) Examples: Mathematical post-processing: E.1, E.2, E.3;

Alternative criteria for P2.d)(vii); type 1 and type 2: E.4

Startup test: E.7;

Tot test: E.5, E.6, E.7;

Online test: E.2, E.7.

D. Evaluation Methodology

Chapter D describes how the evaluator is to examine the specific properties of the respective functionality class. The numbering of the sub-sections begins with i).

D.0 Connection with overall evaluation: The manufacturer specifies the security requirements in the security target. If it is appropriate to specify the generation and use of random numbers in individual cases at this level of abstraction, then functionality class P1 or P2 is specified for the physical random number generator with reference to the security function of the (overall) TOE. Often, the physical random number generator is simply one part of the product to be evaluated. The assumptions for the operational environment and secure use of the TOE must be named.

For common criteria evaluations, you can

- use functionality class FIA_SOS from CC part 2 if the random number generator is used to generate authentication information or
- use family FCS_RND defined in addition to part 2 of the CC.

FIA_SOS.2 TSF generation of secrets

Family behaviour

This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric.

Component levelling

FIA_SOS.2 TSF generation of secrets requires the TSFs to be able to generate secrets that meet the defined quality metric.

Management: FIA_SOS.2

The following actions could be considered for the management functions in FMT:

a) Management of the metrics used to generate the secrets.

Audit: FIA_SOS.1, FIA_SOS.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: Rejection by the TSF of any tested secret;.

b) Basic: Rejection or acceptance by the TSF of any tested secret;

c) Detailed: Identification of any changes to the defined quality metrics.

FIA_SOS.2 TSF Generation of secrets

Hierarchical to: No other components.

FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet [assignment: a *defined quality metric*].

FIA_SOS.2.2 The TSF must be able to enforce the use of TSF generated secrets for [assignment: *list of TSF functions*].

Dependencies: No dependencies.

FCS_RND generation of random numbers

Family behaviour

This family defines quality metrics for generating random numbers intended for cryptographic purposes.

Component levelling

FCS_RND.1 The generation of random numbers using TSFs requires the random numbers to meet the defined quality metrics.

Management: FCS_RND.1

No management functions are provided for.

Logging: FCS_RND.1

There are no events identified that should be auditable if FCS_RND generation of random numbers data generation is included in the PP/ST.

FCS_RND.1 Quality metrics for random numbers

Is hierarchical to: no other components.

FCS_RND.1.1 The TSFs shall provide a mechanism for generating random numbers that meet [assignment: *a defined quality metric*].

FCS_RND.1.2 The TSFs shall be able to enforce the use of TSF-generated random numbers for [assignment: *list of TSF functions*].

Dependencies: FPT_TST.1 TSF testing.

Comment: FCS_RND.1 has been defined in addition to CC part 2 in order to avoid being restricted to class FIA: Identification and authentication when using random numbers and to explicitly describe the usage of random numbers for key generation (FCS_CKM.1) or in cryptographic algorithms or protocols (FCS_COP.1). The desired functionality class P1 or P2 must be compatible with the assignment *List of TSFs* in FIA_SOS.2.2 or FCS_RND.1.2 (comparable to the above aims and possible applications). In addition, FCS_RND.1 provides the connection to start-up tests and online tests for the random number generator.

If the random number generator has a TSF interface visible to the user or, as an interface for a sub-system, determines the TSF behaviour at a higher level design, then

these interfaces must be described in ADV_FSP or ADV_HLD. In the detailed design or low-level design, the focus is on the description of the random number generator as a security mechanism, usage of the random number generator for security-specific functions and interaction with other security mechanisms. A TRNG is to be viewed as a basic component that is able to meet the requirements of ITSEC E4.8 or CC ADV_LLD to be well-defined, largely independent basic components that facilitate testing and minimise the potential for violations of security. The description of the structure, the mode of functioning and the internal interfaces of the random number generator is contained in the detailed design or the low-level design. The internal interfaces defined here must make it clear whether and, where appropriate, how the noise signal, the digitised noise signal and the internal random number sequence can be output. In individual cases, for example in the case of a hardware evaluation of a chip for smart cards, the online test can also be implemented in the software using the smart card operating system. For example, the online test could then be provided by the applicant as a firmware component of the TOE and integrated by the manufacturer of the smart card operating system.

Testing security functions that use TRNGs usually requires statistical tests that can go beyond the test depth specified by ATE_DPT for all TSFs. This proof must be provided according to the desired functionality class for

(1) ITSEC for the effectiveness criteria – Construction, aspect 3 – Strength of mechanisms and as of E2 for Correctness – The development process, phase 4 – Implementation by the manufacturer

(2) CC for EAL1 insofar as identified through the security target as a security function in the context of independent testing (ATE_IND) of the evaluator and from EAL2 for the strength of TOE security functions (AVA_SOF) and functional tests (ATE_FUN) by the manufacturer

Tests of the TRNG also result as tests of the security functions at the level of the detailed design (from ITSEC E3) or ATE_DPT.2 Testing: low-level design (from CC EAL5). Statistical tests of the TRNG must take into account any dependencies of the noise source on the environmental conditions and ageing. Depending on the noise source, therefore, statistical tests must be performed in the permissible conditions of the outer interface power supply and clock provision (e.g. smart cards) as well as the temperature and age (e.g. following artificial ageing). The online tests required if the strength of mechanisms or functions is “medium” or “high” and the tests required for P2, if the strength of mechanisms or functions is “high”, to be performed on the statistical properties of the digitised noise signals during operation are also tested under these conditions.

The analysis of the strength of mechanisms (ITSEC) or strength of functions (CC) must show whether the random numbers deviate from the P1 or P2 properties under certain conditions. This analysis must show whether the effort required by a attacker to transform the TOE into such a state is compatible with the desired strength of mechanisms or functions . If the environmental conditions are hostile, it may be

necessary to extend the tests under the aspect of the strength of mechanisms and the analysis of the weaknesses.

In individual cases, this may influence the delivery documentation, configuration documentation, startup, operation and the guidance documentation.

D.1 Scope and sequence of the evaluation tasks:

Class P1 (continued)

P1.i) Tasks of the evaluator:

The tasks of the evaluator depend on the strength of mechanisms and functions. He must verify requirements P1.d) (i) to (vi) insofar as they are relevant for the desired strength of mechanisms or functions .

P1.i(i) (Testing property P1.d)(i)): The evaluator determines the smallest number c of internal random numbers whose concatenation comprises at least 48 bits. Let $\pi_{1..48}$ be the projection to the 48 bits on the left. Test procedure and decision rule: The evaluator generates internal random numbers r_1, r_2,... and uses these to form a sequence of 2^{16} projections $\pi_{1..48}$ $(r_1,\ldots,r_c)$, $\pi_{1..48}$ $(r_{c+1},\ldots,r_{2c})$,.... He applies disjointness test T0 to this sequence. If this test is passed, property P1.d)(i) is considered to be fulfilled. Otherwise test T0 is applied to a further sequence. If this sequence passes test T0, property P1.d)(i) is considered to be fulfilled. Otherwise it is considered not to be fulfilled. A second repetition is not allowed.

P1.i(ii) (Testing property P1.d)(ii)): Let f be the width of the random numbers generated by TRNG in binary representation and π_w be the projection to the w^{th} component. Sub-sections (ii.a) and (ii.b)(w) describe the ”basic building bocks”, i.e. the individual tests, while (ii.c) describes the entire test procedure including the decision rule.

P1.i(ii.a) The evaluator generates random numbers $r_1,r_2,\ldots$ and interprets them as bit strings with a constant length. He applies tests T1-T4 to the first 20,000 bits in this sequence, as described in Chapter F, with the specified rejection limits. In addition, it calculates test variables $Z_1,\ldots,Z_{5000}$ (see test T5 in chapter F), determines $\max_{\tau\le5000}\{|Z_\tau-2500|\}$ and selects a τ_0 (randomly in case of several candidates) for which this maximum is assumed. He then applies the autocorrelation test (test T5) to the sub-sequence $b'_1 := b_{10001},\ldots,b'_{10000} := b_{20000}$ with shift τ_0 and the rejection limits specified in chapter F.

P1.i(ii.b) (w) ($1 \le w \le f$.) The evaluator generates random numbers $r_1,r_2,\ldots,r_{20000}$. He applies the statistical tests T1-T4 to the sequence of projections $\pi_w(r_1),\ldots,\pi_w(r_{20000})$, as described in Chapter F, with the specified rejection limits. In addition, he calculates test variables $Z_1,\ldots,Z_{5000}$ (see test T5 in chapter F), determines $\max_{\tau \le 5000}\{|Z_\tau - 2500|\}$ and selects a τ_0 (randomly in case of several candidates) for which this maximum is assumed. He then applies the autocorrelation test (test T5) to the sub-sequence $b'_1 := b_{10001},\ldots,b'_{10000} := b_{20000}$ with shift τ_0 and the rejection limits specified in chapter F.

P1.i(ii.c) Test procedure and decision rule: The evaluator successively implements test procedures (ii.a), (ii.b)(1), (ii.b)(2),…, (ii.b)(f), (ii.a),… until a total of 257 bit sequences have been generated and tested. Property P1.d)(ii) is considered to be fulfilled if all the individual tests were passed. If more than one individual test led to rejection, property d)(ii) is considered not to be fulfilled.

If precisely one individual test led to rejection, the entire test procedure must be repeated. Property d)(ii) is considered to be fulfilled if all the individual tests are passed during the repeat procedure. A second repetition is not allowed.

P1.i(iii) (Proof of property P1.d)(iii)): Verification of P1.f)(iv)

P1.i(iv) (Proof of property P1.d)(iv)): Verification of P1.f)(v)

P1.i(v) (Testing property P1.d)(v)): The external conditions (presented/described/explained in the security specifications and the architectural design) must be taken into account insofar as these can influence the function of the noise source. The test procedures and decision rules described in (i) and (ii) must be used for each of these external conditions. Property P1.d)(v) is considered to be fulfilled if properties P1.d)(i) and P1.d)(ii) are fulfilled for all external conditions (possibly proved by means of a test below the limits for the external conditions).

The TRNG is confirmed to belong to class P1 with strength of mechanisms or functions low, strength of mechanisms medium or strength of mechanisms high if P1.d)(i) or P1.d)(i) and P1.d)(ii) or P1.d)(i), P1.d)(ii) and P1.d)(iii) are fulfilled.

If the strength of mechanisms or functions is high, the evaluation tests are performed under different external conditions (temperature, climate, artificial ageing process) and – at least upon external triggering – the TRNG tests the statistical properties of the internal random numbers during operation.

P1.i(vi) (Proof of property P1.d)(vi)): Verification of P1.f)(vi)

P1.j) Explanations of i):

On P1.i)(i): In contrast to the corresponding property K1.d)(i) in [AIS20], the applicant does not have to specify any individual parameters. The reason for this difference is that an acceptable random number generator – unlike a deterministic random number generator – constantly adds entropy to the internal random number sequence. The "quality" of the disjointness properties of the internal random vectors should therefore hardly depend on the width of the selected random number vectors. The probability of an ideal noise generator not being attributed property P1.d)(i) is around 2^{-34}.

On P1.i)(ii): The probability of an ideal noise generator not being attributed property P1.d)(ii) is around $2.5 \cdot 10^{-6}$. (If f=1, then P1.i)(ii.a) trivially coincides with P1.i)(ii.b).)

Class P2 (continued)

P2.i) Tasks of the evaluator:

The tasks of the evaluator depend on the strength of mechanisms and functions. He must verify requirements P1.d) (vii) to (xiii) insofar as they are relevant for the desired strength of mechanisms or functions.

P2.i)(i) Verification of the P1 properties (see P1.i)): Insofar as these are relevant for the desired strength of mechanisms or functions and are not contained in P2-specific requirements.

P2.i)(vii) Verification of property P2.d)(vii): Let k be the width of the binary representation of the digitised noise signals. For k = 1, the following describes five individual tests and formulates a decision rule. For k > 1 the applicant must specify alternative tests where required. Their effectiveness must be justified. These alternative tests shall not be weaker than in the case k = 1.

P2.i)(vii.a) [k=1]: The evaluator generates a digitised noise signal sequence $w_1,\ldots,w_{n0}$ with $n_0 := 100000$. Let $\mu\_emp = (\mu\_emp(0), \mu\_emp(1))$ be its empirical distribution. Property (vii.a) is fulfilled if $|\mu\_emp(1) - 0.5| < a_0 := 0.025$.

P2.i)(vii.b) [k=1]: The evaluator generates a further digitised noise signal sequence $w_1,w_2,\ldots$ which he splits into 2 disjoint sub-sequences $TF_{(0)},\ldots,TF_{(1)}$. Here, the tuple (w_{2j+1},w_{2j+2}) belongs to sub-sequence $TF_{(r)}$ if and only if $w_{2j+1} = r$. The initial sequence $w_1,w_2,\ldots$ must be sufficiently long that both sub-

sequences contain at least $n_1 := 100000$ elements. If we project the first n_1 2-tuple of sub-sequence $TF_{(r)}$ onto the second component, we obtain the one-dimensional sample $St_{(r)}$. If we divide the frequencies at which individual values are assumed by the size of the sample n_1, we obtain the empirical 1-step transition distribution $\nu\_emp_{(r)}$ (.) for predecessor r. Property (vii.b) is fulfilled if $|\nu\_emp_{(0)}(1) + \nu\_emp_{(1)}(0) - 1 | < a_1 := 0.02$.

P2.i)(vii.c) [k=1]: The evaluator generates a further digitised noise signal sequence $w_1,w_2,\ldots$ which he splits into $2^2 = 4$ disjoint sub-sequences $TF_{((0)-(0)))},\ldots,TF_{((1)-(1))}$. Here, the triple $(w_{3j+1}, w_{3j+2}, w_{3j+3})$ belongs to sub-sequence $TF_{((r)-(s))}$ if and only if $(w_{3j+1}, w_{3j+2}) = (r,s)$. The initial sequence $w_1,w_2,\ldots$ must be sufficiently long that each of these four sub-sequences contains at least $n_2 := 100000$ elements. If we project each of the first n_2 3-tuples of sub-sequence $TF_{((r)-(s))}$ onto the third component, we obtain the one-dimensional sample $St_{((r)-(s))}$. For each $s \in \{0,1\}$ the evaluator compares the underlying distributions of the two samples $St_{((0)-(s))}$ and $St_{((1)-(s))}$ with test T7 at the significance level $\alpha\_2:=0.0001$ for equality. Property (vii.c) is fulfilled if both tests are passed. Otherwise property (vii.c) is considered not to be fulfilled.

P2.i)(vii.d) [k=1]: The evaluator generates a further digitised noise signal sequence $w_1,w_2,\ldots$ which he splits into 8 disjoint sub-sequences $TF_{((0)-(0)-(0))},\ldots,TF_{((1)-(1)-(1))}$. Here, the quadruple $(w_{4j+1}, w_{4j+2}, w_{4j+3}, w_{4j+4})$ belongs to sub-sequence $TF_{((r)-(s)-(t))}$ if and only if $(w_{4j+1}, w_{4j+2}, w_{4j+3}) = (r,s,t)$. The initial sequence $w_1,w_2,\ldots$ must be sufficiently long that each of these eight sub-sequences contains at least $n_3 := 100000$ elements. If we project each of the first n_3 quadruples of sub-sequence $TF_{((r)-(s)-(t))}$ onto the fourth component, we obtain the one-dimensional sample $St_{((r)-(s)-(t))}$. For each pair $(s,t) \in \{0,1\}^2$ the evaluator compares the underlying distributions of the two samples $St_{((0)-(s)-(t))}$ and $St_{((1)-(s)-(t))}$ with test T7 at the significance level $\alpha\_3:=0.0001$ for equality. Property (vii.d) is fulfilled if all four tests are passed. Otherwise property (vii.d) is considered not to be fulfilled.

P2.i)(vii.e) The evaluator generates a further digitised noise signal sequence $w_1,w_2,\ldots$ and applies to it the entropy test (test T8) with the parameters L=8, Q=2560 and K= 256000. Property (vii.e) is fulfilled if the test variable $f > 7.976$.

Decision rule: If properties P2.i)(vii.a) - (vii.e) are fulfilled, then property P2.d)(vii) is considered to be fulfilled. If more than one sub-property is not fulfilled, then property P2.d)(vii) is considered not to be fulfilled. If precisely one sub-property is not fulfilled, P2.i)(vii.a) - (vii.e) are applied to another sample. If all sub-properties P2.i)(vii.a) - (vii.e) are fulfilled upon repetition, then property P2.d)(vii) is considered to be fulfilled. A further repetition is not allowed.

If the applicant provides alternative proof in accordance with P2.d) ”Alternative criteria for P2.d)(vii); type 1” and ”Alternative criteria for P2.d)(vii); type 2”, then the evaluator must test this.

P2.i)(vii) (Proof of property P2.d)(viii)): Verification of P2.f)(vii)

P2.i)(viii) (Proof of property P2.d)(ix)): Verification of P2.f)(viii)

P2.i)(ix) (Proof of property P2.d)(x)): Verification of P2.f)(ix)

P2.i)(x) (Proof of property P2.d)(xi)): Verification of P2.f)(x)

P2.i)(xi) (Testing property P1.d)(xii)): The external conditions (presented/described/explained in the security specifications and the architectural design) must be taken into account insofar as these can influence the function of the noise source. For each of these external conditions the test procedures described in (vii) and the decision rules specified there must be used. Property P2.d) (xii) is considered to be fulfilled if properties P2.d)(vii) are fulfilled for all external conditions.

P2.i)(xii) (Proof of property P2.d)(xiii)): Proof of the independent triggering of the online test must be provided as described under D.0.

P2.j) Explanations of i):

On P2.i)(vii.a): Aim and justification: Comparison of the one-dimensional distribution of the digitised noise signal sequence with the uniform distribution on {0,1}. Any dependencies of predecessors are not explicitly taken into account in (vii.a). If the digitised noise signal sequence is memoryless and stationary, and if its distribution $\mu = (\mu(0), \mu(1))$ satisfies the inequation $|\mu(1)-0.5| < 0.025$, then the average entropy increase per bit amounts to more than 0.998.

Comment: In order to ensure that the empirical probabilities are highly likely to lie within the allowed limits, i.e. that digitised noise signal sequences are highly likely to pass this evaluation criterion, the exact probabilities must be closer to 0.5 than is necessary for the empirical probabilities to pass the test. In fact, the probability of this criterion being met is at least 1-0.00078 provided that $\mu(0)=\text{Prob}(w_j=0)$, $\mu(1)=\text{Prob}(w_j=1) \in [0.5-0.02, 0.5+0.02]$. (The specified probability is for the worst case, namely that $\text{Prob}(w_j=0), \text{Prob}(w_j=1) \in \{0.48,0.52\}$.)

On P2.i)(vii.b): Aim and justification: Comparison of the one-step transitional probabilities of the digitised noise signal sequence for various predecessors whereby certain deviations are tolerated. Any multi-step dependencies of predecessors are not explicitly taken into account by criterion (vii.b). If the digitised noise signal sequence

is stationary with memory length ≤ 1, and if the exact transition probabilities $\nu_{(0)}(1)$ and $\nu_{(1)}(0)$ meet requirement (vii.b) rather than the empirical one-step transition distributions $\nu\_emp_{(0)}(1)$ and $\nu\_emp_{(1)}(0)$, i.e. $|\nu_{(0)}(1) + \nu_{(1)}(0) - 1 | < a_1 := 0.02$, then the average entropy increase per bit is at most 0.00057 less than if the digitised noise signal sequence was memoryless but had the same stationary distribution. Comment: for memoryless noise signal sequences the following applies: $\nu_{(0)}(0)=\nu_{(1)}(0)$ and $\nu_{(0)}(1)= \nu_{(1)}(1)$, i.e. $\nu_{(1)}(0) = 1-\nu_{(0)}(1)$. A small value $|\nu\_emp_{(0)}(1) +\nu\_emp_{(1)}(0) -1 |$ therefore indicates that at best only weak one-step dependencies exist. If the inequation $|\nu_{(0)}(1) +\nu_{(1)}(0) -1 | < 0.012$ is satisfied for the exact one-step transition probabilities, then the probability that the empirical one-step transition probabilities meet criterion (vii.b) is at least 1-0.00017.

On P2.i)(vii.c) and (vii.d): Aim: The digitised noise signal sequences should not have any dependencies higher than one-step ones. If this is the case, then for m > 1 the m-step transition probabilities do not depend on the m^{th} predecessor in particular. In other words, if the first (m-1) predecessors are constant, all m last predecessors induce the same distribution on {0,1}.

On P2.i)(vii) (decision rule): The probability that an ideal TRNG does not fulfil property (vii.a),… or (vii.e) when implemented once is 0, 0, $2\cdot10^{-4}$, $4\cdot10^{-4}$ or 0. The decision rule ensures that the probability of an ideal TRNG incorrectly not being attributed property (vii) is only around $6\cdot10^{-7}$. The explanations for (vii.a), (vii.b) and (vii.e) also show that even TRNGs for which the skewness and the one-step dependencies of the digitised noise signal sequences do not exceed certain limits are highly likely to fulfil the corresponding individual properties. The probabilities of fulfilling properties (vii.c) and (vii.d) are practically identical for all distributions as long as they do not have any multi-step dependencies. In the case of distributions for which $\mu(0)$, $\mu(1)$, $\nu_{(0)}(1)$ and $\nu_{(1)}(0)$ (instead of the empirical values) meet the requirements from (vii.a) and (vii.b), the rejection probability for the entropy test is < 10^{-4}.

On P2.i)(vii) (case k>1): For the case k>1 it appears appropriate to interpret the digitised noise signals (blocks of k bits) as binary-value sub-sequences of length k and to apply criteria (vii.a) – (vii.e) to the entire binary sequence. For such a procedure, however, two fundamental phenomena must be taken into account. Depending on the noise source and the precise form of digitisation, the individual bits of the digitised noise signals do not have the same distribution. Therefore, a stationary digitised noise signal sequence does not necessarily cause a stationary binary-value sequence. Moreover, any one-step dependencies of the digitised noise signal sequence generally induce k-step dependencies in the binary-value sequence.

D.2 Comment: The evaluation of a TRNG is essentially based on statistical tests. The evaluation result cannot therefore be reproduced with certainty. This circumstance is moderated by the fact that it is unlikely for the P1 or P2 property not to be recognised for "reasonable" TRNGs. The result of a TRNG evaluation is thus ”quasi-reproducible”, which, of course, is indispensable for the reliability and trustworthiness of an evaluation procedure.

E. Examples

This chapter describes various mathematical post-processing methods and online test variants as examples and examines them with regard to the corresponding requirements formulated in P1.d) and P2.d). The evaluator must verify or disprove empirically whether requirements P1.d)i), ii) and v) as well as P2.d)vii) and (xii) are met. Properties P1.d)(i) and (ii) are relatively weak and should be fulfilled by practically every physical noise source. As *null hypothesis* we assume that the random numbers to be tested have been generated by an ideal noise source.

E.1 Example: (mathematical post-processing, total failure of the noise source)

Each digitised noise signal comprises one bit. The mathematical post-processing consists of a linear fedback shift register of length 63 with the primitive feedback polynomial $p(x) = x^{63}+x^{31}+1$. The digitisation of the noise signal and the stepping cycle of the shift register are synchronous. The feedback bit is the next internal random number. The XOR sum of the feedback bit with the current digitised noise signal bit is fed back into the shift register.

For each initial assignment, the mathematical post-processing bijectively maps the set of finite digitised noise signal sequences to the set of finite internal random number sequences. The mathematical post-processing thus fulfils in particular property P2.d)(viii).

If the digitised noise signal sequence is constantly 0,0,..., from a point in time, the internal random number sequence could trivially be generated using the free-running linear shift register. This is essentially also the case if the digitised noise signal sequence is constantly 1,1, ... Only the initial assignment of the shift register must be inverted bit by bit, and the output sequence likewise. (This is the case for every linear shift register with an even number of taps.) In the case of total failure of the noise source, the mathematical post-processing in both cases corresponds to a K2-DRNG in accordance with [AIS20] (see also example E.3 in [AIS20]).

Even if the noise source fails completely, the internal random numbers should pass practically all common statistical tests. It is much more efficient, however, to test the digitised noise signal sequence (see P2.d)(vii), (ix), (xi)).

E.2 Example: (mathematical post-processing): Let the digitisation deliver a sequence $X = (x_0, x_1, \ldots)$ of independent bits with probability $P\{x_i = 1\} = p \leq 1/2$.

We will begin with an example from von Neumann. Let the bit stream be divided into pairs $(x_{2i}, x_{2i+1}), i = 0,1,2,\ldots$. The pairs (00) and (11) are discarded and the remaining sequence *y* is transformed into an internal random sequence *y'* in accordance with

$$y'_k = \begin{cases} 0 \ for \ (y_{2k}, y_{2k+1}) = (01) \\ 1 \ for \ (y_{2k}, y_{2k+1}) = (10) \end{cases}$$

The sequence $(y'_0, y'_1, \ldots)$ is then asynchronous to the sequence X, but uniformly distributed.

For further examples, the digitised noise signal sequence is divided into segments $\overline{X}_j = (x_{64j}, x_{64+1}, \ldots, x_{64j+63})$ each of 64 bits and post-worked to form an internal random sequence $\overline{Y} = (Y_0, Y_1, \ldots)$ where $Y_j = (y_{64j}, y_{64j+1}, \ldots, y_{64j+63})$. Three post-processing variants are considered:

a) Y_j corresponds to the left 64 bits of section X_j, padded with a known pattern and then hashed

b) XOR sum of two consecutive segments, $Y_j = X_{2j} \oplus X_{2j+1}$

c) Encryption using a key from the digitised noise signal sequence, $Y_j = E_{X_{2j}}(X_{2j+1})$

First, assume it has been established that each segment of the digitised noise signal sequence assumes a value A with a probability that is determined from its Hamming weight $|A| = \sum_{i=0}^{63} a_i$, i.e. the number of ones, to $P\{\overline{X}_j = A\} = p^{|A|} \cdot (1-p)^{(64-|A|)}$. Let $W = (w_0, w_1, \ldots, w_{2^{64}-1})$ be the non-falling sequence of these probabilities.

The post-processing in variant a) can be interpreted as a random mapping. It is not injective and, in particular, does not fulfil property P2.d)(viii). However, it should be noted that statistical properties of small structures, such as the 0/1 ratio, can be thereby improved. The statistical properties of the segment sequence $(\overline{Y}_j)_{j=0,1,\ldots}$, such as the probability of segment values being repeated, even become slightly worse.

Variants b) and c) compress the digitised noise signal sequence to form an internal random sequence in the ratio 2:1 They have no memory and generate independent segments of the internal random sequence. Variant b) supplies independent bits and

smoothes the skewness of the bit distribution by $p-2p^2$, i.e. $P\{a_i=1\}=2p-2p^2$, $P\{a_i=0\}=1-2p+2p^2$.

For variant c), the identity $P\{\hat{Y}_j=A\}=\sum_{K\in\{0,1\}^{64}} P\{\overline{X}_{2j}=K\}\cdot P\{\overline{X}_{2j+1}=E_K^{-1}(A)\}$ applies for all *j*.

If $E_K^{-1}(A)$ delivers a permutation[2] for A fixed and a running $K\in\{0,1\}^{64}$, these probabilities can be estimated by $p^{64}\le 2^{64}\,p^{64}(1-p)^{64}\le P\{Y_j=A\}\le(1-2p+2p^2)^{64}\le(1-p)^{64}$. For $p<1/2$ the probability distribution is smoothed. However, the bits within a segment are generally not independent, even though these dependencies can be ignored for many applications.

As the bit distribution is skewed, an attacker can sort the set of segments in the digitised noise signal sequence by the probabilities of them occurring. This favours exhaustion attacks against individual segments of the digitised noise signal sequence. For variant a) this renders exhaustion attacks against the internal random number segments at least equally efficient. This is not the case for variants b) and c) because two digitised noise signal segments are used in each internal random number.

Like the digitised noise signal sequence, the internal random numbers, too, are independent. In variant b) an online test of the noise source could be performed in accordance with requirement (xi) by monitoring the 0/1 distribution of the internal random sequence. In variants a) and c) this would be ineffective because the bit distribution is blurred by the hashing or encryption.

E.3 Example: (mathematical post-processing)

The noise source generates individual bits whose distribution can be considered stationary due to the mathematical model of the noise source. Moreover, the assumption that the distribution is stationary has been confirmed by statistical examinations of various prototypes. (To be more precise, the assumption that the distribution is stationary could not be rejected by corresponding statistical tests at a small significance level α (e.g. α=0.001).) The digitised noise signal sequence does not necessarily have to be independent. Furthermore, we have a mapping f: $\{0,1\}^m \times \{0,1\}^m \to \{0,1\}^m$ (e.g. m=8) that is bijective in the second (or first) component if the first component (or second component) is fixed.

The digitised noise signal sequence is segmented into non-overlapping blocks $X_1,X_2,\ldots$ of length m from which the internal random number sequence is generated using $Y_1:=f(X_1,X_2)$, $Y_2:=f(X_3,X_4),\ldots$.

With a simple calculation we can verify that $H(Y_{j+1} \mid Y_1,\ldots,Y_j) \ge H(X_{2j+1} \mid X_1,\ldots,X_{2j})$, i.e. the entropy increase per m-bit block is at least as great for the internal random number

[2] Please note that not all cipher mappings fulfil this property.

sequence as for the digitised noise signal sequence. In general (depending on the distribution!) it will be truly larger. The entropy increase is obtained by halving the throughput. The map f can, for example, be a group operation on $\{0,1\}^m$ (see also example E.2 b) for m = 1).

E.4 Example: (Alternative criteria for P2.d)(vii); type 1 and type 2)

This example looks at various mathematical post-processings to determine whether an evaluation in accordance with ”Alternative criteria for P2.d)(vii); type 1” or ”Alternative criteria for P2.d)(vii); type 2” could be used instead of P2.d)(vii). We assume that the noise source generates binary-value digitised noise signal sequences.

We examine the following mathematical post-processings:

a) Example E.1

b) Non-overlapping, consecutive noise signal bit pairs are XORed (=special case of example E.3 where m=1 and $f(X_1,X_2):= X_1 \oplus X_2$)

c) Example E.2b)

Scenario 1: The digitised noise signal sequence can be accessed. Extensive investigations of prototypes have shown that the probability of a "1" lies in the interval [0.45,0.47]. The statistical behaviour of the digitised noise signals is the same as realisations of independent random variables. In particular, they meet criteria P2.i)(vii.b), P2.i)(vii.c) and P2.i)(vii.d).

On a) Evaluation in accordance with the ”Alternative criteria for P2.d)(vii); type 1” is *not* possible because the mathematical post-processing does not increase the entropy per bit.

On b) As the digitised noise signal sequence is independent, it follows that the internal number sequence is also independent. Assuming that the digitised noise signal is independent, the probability of a ”1” for the internal random numbers lies in the interval [0.49875, 0.50125]. This note provides the comprehensible proof required by the second point of ”Alternative criteria for P2.d)(vii); type 1”.

On c) Statistical tests of the internal random number sequences cannot deliver any useful statements (in the sense of the aims of criterion P2.d)(vii)) because such tests would have to take into account the one-dimensional distribution of entire 64-bit blocks. The necessary sample sizes would then be much higher than 2^{64}, which is practically impossible. For an evaluation in accordance with the ”Alternative criteria for P2.d)(vii); type 1”, therefore, the applicant would have to provide theoretical proof that the mathematical post-processing increases the entropy per bit sufficiently (see P2.j)).

Scenario 2: The digitised noise signal sequence cannot be accessed. Due to the precise technical realisation of the physical noise source (i.e. taking into account the switching times and dead times of individual building blocks, sampling rates, etc.) it is plausible

to assume that the generated digitised noise signal sequences are independent. (The justification for deriving this model assumption is extremely important.)

Comment: In contrast to scenario 1, it may be the case that the digitised noise signal sequence fulfils criterion P2.d)(vii) but no direct proof can be provided.

On a) The internal random numbers can be used to determine the digitised noise signal sequence. Tests P2.i)(vii.a) to P2.i)(vii.e) can be applied to the back-calculated digitised noise signal sequence. If the digitised noise signal sequence does not pass tests P2.i)(vii.a) - P2.i)(vii.e), then evaluation in accordance with the ”Alternative criteria for P2.d)(vii); type 2” is *not* possible.

On b) Since the digitised noise signal sequence is independent, the internal random bit sequence is also independent (see scenario 1). A suitable statistical test, as required by the third point of the ”Alternative criteria for P2.d)(vii); type 2”, is provided by applying test P2.i)(vii.a) to the internal noise signal sequence. (Note that this test has to be implemented anyway due to the first point.)

On c) Statistical tests of the internal random number sequences cannot deliver any useful statements (in the sense of the aims of criterion P2.d)(vii)) because such tests would have to take into account the one-dimensional distribution of entire 64-bit blocks. The necessary sample sizes would then be much higher than 2^{64}, which is practically impossible. An alternative evaluation in accordance with the ”Alternative criteria for P2.d)(vii); type 2” therefore seems hardly possible.

E.5 Example: (tot test, startup test)

The TRNG continuously generates binary-value digitised noise signal sequences. No mathematical post-processing is performed, which means that the digitised noise signals correspond to the internal random numbers. Random numbers are output from a 512-bit FIFO. If the FIFO contains no more than 256-bit random numbers, no further random numbers can be extracted. Rather, the FIFO must first be filled up again. If at least 48 consecutive internal random numbers (bits) are identical, then the TRNG is shut down because it is suspected that the noise source has failed completely. The same test is performed when the TRNG is switched on (startup test).

This test specification fulfils T1.d(iii), T1.d(iv) and T2.d(x) because no random numbers can be output that are generated following total failure of the noise source. However, property P2.d)(ix) is not fulfilled because this test does not even detect extremely obvious statistical weaknesses. In particular, the test is not a useful online test.

E.6 Example: (online test, tot test)

The TRNG continuously generates internal random numbers. Random numbers are output from a 512-bit FIFO. When triggered externally, the binary representations of consecutive internal random numbers are interpreted as bit strings. If the FIFO is half

empty, it is filled up using currently generated consecutive internal random numbers $r_1,r_2,....$ All internal random numbers used to fill up the FIFO are tested. In order to be tested, they are interpreted as bit strings and segmented into 4-bit words. A χ^2 modification test is applied to each 80 (4-bit words) (see, for example, [Ka], 69). (Note that these random numbers do not have to be kept available outside the FIFO. Rather, it is sufficient to maintain 16 word frequency counters internally.) The null hypothesis is rejected if the test variable is > 65.0. According to ([Ka], 69), the test variable is approximately χ^2-distributed with 15 degrees of freedom, which gives rise to the significance level $3.8 \cdot 10^{-7}$ (see table 1 in example E.7). If this test leads to the null hypothesis being rejected, the TRNG is shut down and a suitable error message is generated. The error message is logged and the TRNG must be restarted manually. Internal random numbers that are not used to fill up the FIFO are neither saved nor tested. It is to be expected that the TRNG calls the online test around 1000 times each year. For ideal random number generators, the χ^2-distribution function gives rise to a probability of around $3.8 \cdot 10^{-4}$ that there will be at least one noise alarm each year. Requirement P1.d)(vi) is therefore met.

Whether this online test detects total failure of the noise source depends on the mathematical post-processing. For the post-processing in example E.1, total failure is not detected. In this case, however, the TRNG behaves like a K2-DRNG as defined by [AIS20] (see P1.d)(iv) even following total failure of the noise source.

E.7 Example: (online test, tot test, startup test)

As in example 4.6, the internal random numbers are written to a 512-bit FIFO that is filled up using currently generated consecutive internal random numbers $r_1,r_2,...$ as soon as it is at least half empty, and at the latest when only 128 bits are left in the FIFO. All digitised noise signal bits from which the internal random numbers used to fill up the FIFO are generated are tested in an online test (”basic test”; see below). If the FIFO is full again but the sample for applying a basic test is not yet complete, then the sample is filled up using the subsequently generated digitised noise signal bits and the test is evaluated. Only then can internal random numbers be output again. In addition, the TRNG independently performs a further online test each minute.

The startup test performed when the TRNG is started consists of one single χ^2 test over 128 (4-bit words). The TRNG passes the startup test if the test variable is ≤ 65.0 (see also example E.6). It is the task of the startup test to ensure that the noise source is functioning properly and detect any very obvious statistical weaknesses. The startup test thus fulfils property P2.d)(ix). It is left to the online tests to reveal less obvious statistical weaknesses (see also [Sch]).

To begin with, the basic test type is determined. The mathematical model of the noise source should be taken into account here because an unsuitable basic test may lead to a considerable reduction in the effectiveness of the online test. (The following section

does not cover selecting basi tests in more detail.) In this example, the basic test is a χ^2 test over 128 (4-bit words).

A *test suite* is made up of a maximum of N=512 basic tests (= χ^2 tests over 128 (4-bit words)). In the following we will use $C_1,C_2,...$ to refer to the test variables for the basic tests. Moreover, $H_0:=EW_0(C_1)$ (=expected value of the basic test variable under the null hypothesis) and $H_j:=(1-\beta)H_{j-1}+\beta C_j$ for $j\geq 1$ with $\beta=2^{-6}$, whereby test variables C_j and H_j are each rounded to 6 binary place bits. (In particular, this makes it possible to calculate the "history variables" $H_1,H_2,...$ using integer arithmetic.) The following two evaluation rules apply for each step $1 \leq j \leq N$:

(i) If C_{j-2}, C_{j-1} ,$C_j > 26.75$, then there is a preliminary noise alarm

(ii) If $H_j \notin [13.0,17.0]$, then there is a preliminary noise alarm

If no preliminary noise alarm occurs within a test suite, a new test suite is started following 512 basic tests. Each preliminary noise alarm causes the current test suite to be cancelled and the FIFO to be deleted. The preliminary noise alarm is logged. If three consecutive test suites are stopped due to a preliminary noise alarm, a noise alarm occurs, the TRNG is shut down and a corresponding error message is generated.

For reasons of simplicity, we will assume in the following that the digitised noise signals are independent. To be precise:

Distribution assumption for the digitised noise signal sequence (in example E.7): Due to the mathematical model of the noise source and statistical investigations of prototypes, the distribution of the digitised noise signal bits can be considered to be stationary and independent. No dependencies on predecessors could be determined and the prototypes investigated met requirement P2.d)(vii). The digitised noise signal sequence can thus be seen as the realisation of independent random variables whereby the probability $\mu(1)$ of the value "1" may depend on the individual device and may change in the course of time (ageing effects).

Specifications (for example E.7; see also P2.d)(xi)): For the intended applications it is fully sufficient if $\mu(1)\in[0.49,0.51]$. If this probability lies outside of the interval [0.475,0.525], the online tests should soon recognise this and trigger a noise alarm.

Table 1 shows the probabilities of a preliminary noise alarm within a test suite and the average number of noise alarms per year. Here it has been assumed that 1584 basic tests are performed each day (of which 144 are based on the event of filling up the FIFO).

Table 1

$\mu(1)$	Probability of a noise alarm within a test suite	Average number of noise alarms per year
0.500	0.0162	0.0047
0.495 or 0.505	0.0187	0.0072
0.490 or 0.510	0.0292	0.027
0.485 or 0.515	0.0794	0.52
0.480 or 0.520	0.2954	21.1
0.475 or 0.525	0.7670	
0.470 or 0.530	0.9912	

Comparison with the online test from example 4.6: In that example, a noise alarm occurs if a single test delivers a value greater than 65.0. (Moreover, the internal random numbers were tested in example 4.6.) This is an event that occurs very seldom, at least under the null hypothesis (= independent and uniformly distributed 4-bit words). However, such an approach has disadvantages: on the one hand, even under the null hypothesis, test variable distribution is normally only asymptotic. i.e. the limit distribution for a sample size tending towards infinity (= χ^2 distribution with 15 degrees of freedom). If the sample size is small, the relative error $| p_{exact} - p_{approx.} | / p_{approx.}$ for large rejection bounds can be large. (Here, p_{exact} is the exact rejection probability, whereas $p_{approx.}$ is the approximate rejection probability calculated from the χ^2 distribution.) As a result, the number of noise alarms is considerably greater than is to be expected based on asymptotic limit distribution. If only a small leeway is allowed, this leads to an overly large failure or shut-down rate. (For example, the relative error for a sample size of 80 (4-bit words) for the rejection limit 65.0 is 10.1. The online test from E.6 meets the requirements of P1.d). However, it nonetheless appears appropriate to increase the sample size of the sample.) If a large leeway is allowed, on the other hand, this can have the result that unacceptable weaknesses are recognised very late, if at all. Furthermore, it is hardly possible to make statements about the rejection probability if the distribution on which the sample is based deviates from the null hypothesis.

For the online test procedure proposed in example 4.7, the situation is more favourable: Under the null hypothesis, $\text{Prob}(C_j > 26.75) \approx 0.03$. Here, the χ^2 distribution still has "weight" and the relative error is low. The decision rule (ii), too, does not depend on the occurrence of a single, very rare event, but on a sequence of numerous events that taken individually are not in the least bit rare. The small weighting factor β ensures this.

If the distribution of the digitised noise sequence deviates from the null hypothesis (independent and uniformly distributed), the distribution function of the test variable

can be approximated by means of stochastic simulation (see, for example, [Dev]). Here, a pseudo-random number generator (for example with a linear congruence generator or a linear shift register; unpredictability properties of the pseudo-random numbers are irrelevant here) is used to generate standard random numbers, i.e. pseudo-random numbers that are uniformly distributed on the interval [0,1]. From this we derive a long bit sequence (e.g. (4*128)*1000000 bits) in accordance with the desired distribution, segment this sequence into sub-sequences of length 512 bits and apply a χ^2 test to each segment. For the distributions taken into account in table 1, stochastic simulations deliver the following probabilities that the test variable is >26.75: 0.0299 (null hypothesis), 0.0303, 0.0331, 0.0371, 0.0416, 0.0526 and 0.0656 (order as in table 1).

The basic test variables $C_1, C_2, \ldots$ can be interpreted as the realisation of independent random variables. Decision rules (i) and (ii) define a homogenous Markov chain on the finite state space $\Omega = \{(2^{-6}k, i) \mid k \in \mathbb{N}, 2^{-6}k \in [13.0,17.0], 0 \leq i \leq 2\} \cup \{\omega\}$, where ω is an absorbing state. The state (v,i) is reached if the history variable assumes the value v and the last $i \leq 2$ test variables were greater than 26.75. The absorbing state ω is reached if a noise alarm is triggered (see also [Sch]).

The online test meets requirements P2.d)(xi) and (xiii), and – provided that the mathematical post-processing meets requirement P2.d)(viii) – they also meet requirement P1.d)(vi).

For each value H_{j-1}, $C_j \geq 269.5$ always triggers a noise alarm due to decision rule (ii). In particular, this is guaranteed if the last 220 bits of a sample are constantly 0 or constantly 1. Following total failure of the noise source, the current basic test does not necessarily lead to a noise alarm. However, a noise alarm is triggered at the latest by the subsequent basic test. At this point in time, however, no internal random number that has been used to fill up the FIFO after the total failure occurred has left the FIFO. Following two further online tests, the TRNG is shut down without outputting any further internal random numbers beforehand. Requirement P2.d)(x) is thus also fulfilled.

F. Statistical tests

The following lists the statistical tests needed to verify the P1-specific properties P1.d)(i), (ii) and (v) as well as the P2-specific properties P2.d)(vii) and (xii).

F.1 Comment:

(i) Tests T0 to T5 are applied to internal random numbers (see P1.i)). Assuming that sequences $w_1,\ldots,w_{2\wedge16}$ and $b_1,\ldots,b_{20000}$ are generated by ideal noise sources, the following rejection probabilities result: Test T0: 2^{-17}, tests T1 to T5: Each 10^{-6}.

(ii) Tests T6 to T8 are applied to digitised noise signal sequences (see P1.i)). Assuming that sequences $w_1,\ldots,w_n$ and $b_1,\ldots, b_{(Q+K)L}$ are generated by ideal noise sources, the rejection probabilities are negligible for the parameters selected in P2.i). As digitised noise sequences from real TRNGs usually display statistical defects (skewness, dependencies), the rejection limits are selected in such a way that TRNGs with tolerable weaknesses pass these tests (see P2.j)).

(iii) Tests T1 – T4 are taken from the document [FI140-1] (4.11.1), together with their names and rejection limits. In order to prevent possible confusion, it should be noted that although [FI140-2] also describes tests T1-T4 the rejection limits are different. Assuming that sequences $b_1,\ldots,b_{20000}$ are generated by an ideal noise source, the rejection probabilities in [FI140-2] are 10^{-4} per test.

(iv) The theoretical background to the entropy test (test T8) is described in [Cor].

Test T0 (disjointness test)

The sequence $w_1,\ldots,w_{2\wedge 16} \in \{0,1\}^{48}$ passes the disjointness test if the subsequent members are pairwise different.

Test T1 (monobit test)

$$X = \sum_{j=1}^{20000} b_j$$

The bit sequence $b_1,\ldots,b_{20000}$ passes the monobit test if $9654 < X < 10346$.

Test T2 (poker test)

For $j = 1,\ldots,5000$ let $c_j = 8\cdot b_{4j-3}+4\cdot b_{4j-2}+2\cdot b_{4j-1}+b_{4j}$. Furthermore, $f[i] := |\{j: c_j=i\}|$.

$$Y = (16/5000)\cdot\left(\sum_{i=0}^{15} f[i]^2\right) - 5000$$

The bit sequence $b_1,\ldots,b_{20000}$ passes the poker test ($=\chi^2$ modification test with 15 degrees of freedom) if $1.03 < Y < 57.4$.

Test T3 (run test)

A run is a maximum sub-sequence of consecutive zeroes or ones.

The bit sequence $b_1,\ldots,b_{20000}$ passes the run test if the number of occurring run lengths lies within the permitted intervals, as specified below. The runs of zeroes and ones are evaluated separately.

Run length	Permitted interval
1	2267-2733
2	1079-1421
3	502-748
4	233-402
5	90-223
≥ 6	90-233

Test T4 (long run test)

A run of length ≥ 34 is called a long run.

The bit sequence $b_1,\ldots,b_{20000}$ passes the long run test if no long run occurs.

Test T5 (autocorrelation test)

For $\tau \in \{1,\ldots,5000\}$, $Z_t = \sum_{j=1}^{5000} (b_j \oplus b_{j+t})$.

The bit sequence $b_1,\ldots,b_{20000}$ passes the autocorrelation test (with shift τ) if $2326 < Z_t < 2674$. (Please note that the sub-sequence $b_{10001},\ldots,b_{20000}$ is not used in the test variable.)

Test T6 (uniform distribution test)

The sequence $w_1,\ldots,w_n \in \{0,1\}^k$ passes the uniform distribution test with parameters (k,n,a) if:

(*) $\frac{1}{n} \cdot \left| j \le n \mid w_j = x \right| \in [2^{-k} - a, 2^{-k} + a]$ for all $x \in \{0,1\}^k$.

Comment: for k=1 the condition (*) is simplified to $\frac{1}{n} \cdot \left| j \le n \mid w_j = 1 \right| \in [2^{-k} - 0{,}5, 2^{-k} + 0{,}5]$. If in addition n = 20000 and a = 0.0173, then the uniform distribution test corresponds to the monobit test T1.

Test T7 (comparative test for multinomial distributions)

For each i ∈ {1,…h} let the n-element sample $w_{i1},\ldots,w_{i,n}$ assume values from the set {0,1,…,s-1}. According to the null hypothesis, the multinomial distributions on which the individual samples are based are identical. Furthermore, for t ∈ {0,…,s-1} let $f_i[t]:=|\{j: w_{ij}=t\}|$, and let $p_t:= (f_1[t]+\ldots+ f_h[t])/(hn)$ be the relative frequency for the occurrence of t determined from the total of all samples. Under the null hypothesis, the test variable $\sum_{i=1,..,h}\sum_{t=0,\ldots,s-1}(f_i[t]-np_t)^2/np_t$ is approximately χ^2-distibuted with (h-1)(s-1) degrees of freedom ([Ka], Test 76). In the special case where h = s =2 and the significance level α = 0.0001 (see P2.i)(vii.c) and P2.i)(vii.d)), the rejection limit is 15.13.

Test T8 (entropy test)

The entropy test is performed in accordance with Coron [Cor]. The bit sequence $b_1,\ldots,b_{(Q+K)L}$ is segmented into non-overlapping output words $w_1,\ldots,w_{Q+K}$ of length L. A_n is the distance from w_n to its predecessor with the same value, and

$$A_n=\begin{cases} n & \text{if no } i<n \text{ existwith } w_n=w_{n-i} \\ \min\{i \mid i\geq 1,\ w_n=w_{n-i}\} & \text{in all other cases}\end{cases}$$

- Test variable f: $\{0,1\}^{(Q+K)L}\rightarrow$R is determined for the Coron test by

$$f_C(\overline{s})=\frac{1}{K}\sum_{n=Q+1}^{Q+K}g(A_n) \text{ where } g(i)=\frac{1}{\log(2)}\sum_{k=1}^{i-1}\frac{1}{k}.$$

For $i\geq 23$, we can estimate the total in the function $g(i)$ with an error under 10^{-8} by:

$$\sum_{j=1}^{n}\frac{1}{j}=\log n+g+\frac{1}{2n}+\frac{1}{12n^2}+\mathrm{O}\left(\frac{1}{n^4}\right),\ g\approx 0.577216 \text{ (EULER constant)}$$

For a stationary binary-value random source with a finite memory, the expected value for test variable f_C is closely related to the entropy increase per L-bit block. Indeed, if the noise source is independent, the two are equal. For ideal noise sources, a good approximation of the distribution of test variable f_C is provided by a normal distribution with expected value μ_C and variance $(\sigma_C)^2$.

$$s_C=c_C(L,K)\sqrt{Var\left(g(A_n)\right)/K},\quad c_C(L,K)=d(L)+\frac{e(L)\cdot 2^L}{K}$$

Table 2 (the values are valid for ideal noise sources ([Cor]))

L	Variance Var($g(A_n)$)	d(L)	e(L)
3	2.5769918	0.3313257	0.4381809
4	2.9191004	0.3516506	0.4050170
5	3.1291382	0.3660832	0.3856668
6	3.2547450	0.3758725	0.3743782
7	3.3282150	0.3822459	0.3678269
8	3.3704039	0.3862500	0.3640569
9	3.3942629	0.3886906	0.3619091
10	3.4075860	0.3901408	0.3606982
11	3.4149476	0.3909846	0.3600222
12	3.4189794	0.3914671	0.3596484
13	3.4211711	0.3917390	0.3594433
14	3.4223549	0.3918905	0.3593316
15	3.4229908	0.3919740	0.3592712
16	3.4233308	0.3920198	0.3592384
infinite	3.4237147	0.3920729	0.3592016

Example: for L=8 and K=256,000, $\sigma_C \approx 0.0014$.

In contrast to the Maurer test ([Mau], [CoNa]), the Coron test delivers more than asymptotic entropy statements, at least for independent random sequences. For the Maurer test, an implementation by the NIST is available at http://csrc.nist.gov/rng/ [STS].

G. Literature

[AIS20] AIS 20 (Version 1 of 02.12.99): Functionality classes and evaluation methodology for deterministic random number generators .

[CC] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; May 1999 and ISO 15408-2:1999.

[Cor] J.-S. Coron: On the Security of Random Sources, Gemplus' Corporate Product R&D Division, Technical Report IT02-1998.

Also in: H. Imai and Y. Zheng (eds.): Public Key Cryptography. Second International Workshop on Practice and Theory in Public Key Cryptography, PKC'99. Springer, Lecture Notes in Computer Science 1560, Berlin 1999, 29-42.

[CoNa] J.-S. Coron and D. Naccache: An Accurate Evaluation of Maurer's Universal Test. In: S. Tavares and H. Meijer (eds.): Selected Areas in Cryptography '98, SAC '98. Springer, Lecture Notes in Computer Science, Vol 1556, Berlin 1999, 57-71.

[FI140-1] FIPS PUB 140-1 (January 11, 1994), NIST, Security Requirements for Cryptographic Modules.

[FI140-2] FIPS PUB 140-2 1999, NIST, Security Requirements for Cryptographic Modules.

[FI186] FIPS PUB 186-1 (December 15, 1998), NIST, Specifications for the Digital Signature Standard (DSS).

[ITSEC] Information Technology Security Evaluation Criteria (ITSEC); Provisional Harmonised Criteria, Version 1.2, June 1991.

[ITSEM] Information Technology Security Evaluation Manual (ITSEM); Provisional Harmonised Methodology, Version 1.0, September 1993.

[JIL] Information Technology Security Evaluation Criteria ITSEC Joint Interpretation Library (ITSEC JIL), Version 2.0, November 1998.

[Ka] G.K. Kanji: 100 Statistical Tests. Sage Publications, London 1995.

[Mau] U. Maurer: A Universal Statistical Test for Random Bit Generators. J. Cryptology (1992), 89-105.

[RSA] PKCS#1: RSA Encryption Standard. An RSA Laboratories Technical Note, Version 1.5, November 1, 1993.

[Sch] W. Schindler: Efficient Online Tests for True Random Number Generators. Appears in: C.K. Koc, D. Naccache, C. Paar (eds.): Cryptographic Hardware and Embedded Systems – CHES 2001, Springer, Lecture Notes in Computer Science, Vol. 2162, Berlin 2001.

[STS] A Statistical Test Suite for Random and Pseudorandom Numbers. NIST Special Publication 800-22 (December 2000).